DEVELOPING PEER to PEER MULTIMEDIA ALGORITHM for LIVE STREAMING to ENHANCE QOS IN CLOUD ENVIRONMENT

A RESEARCH SYNOPSIS

SUBMITTED TO

SRTM UNIVERSITY

SCHOOL OF COMPUTATIONAL SCIENCES

SUBMITTED BY

SHARF ALDIN MUSA IDRIS ELNUOR

Abstract

The two most popular architectures in the modern world are Peer-to-Peer Systems and Cloud Systems. Peer-to-Peer Systems (P2P) have become very popular in recent times, mainly being used to facilitate file sharing among disparate systems. Another recent trend in modern computing has been the wide scale utilization of Cloud Computing architectures. These systems are used to allow multiple systems to pool their resources and allow other tertiary systems to use these shared resources in bulk for tasks such as data storage, complex calculations, and file sharing. This entails the conceptual outsourcing of various data processing tasks to an external cloud system. Given the two nearly independent functionalities of P2P and Cloud architectures, it is interesting to consider the possibility of fusing these two concepts and researching the applications of the resultant amalgamation

Multimedia streaming remains as an ever growing field where recent researches are being carried out in improving the Quality of streaming services. Peer to Peer live media streaming is a flourishing technology that thrives to turn down the blockades in streaming live events over

the Internet. However, guaranteeing an outrageous Quality of Service using P2P overlay network raises a number of major challenges: (i) Guaranteeing QoS in the presence of peer Churn/Flash crowd, (ii) Exaggerate peers to participate in content distribution, (iii) Elimination of bandwidth bottleneck in the overlay, (iv) Network Address Traversal based Peer identification and (v) Handling of Device Heterogeneity. The existing P2P streaming techniques solved issues but found to be inappropriate especially for live streaming. The potential solution for the above problem is assisting the P2P streaming network with a cloud computing infrastructure to guarantee the desired level of QoS.

In this research, we try to discuss the theory and application of Cloud Based Peer-to-Peer Systems and their potential application in multimedia streaming services. While the value of P2P Systems and Cloud Computing Systems have been extolled individually, the hybrid of both concepts shows great promise. In this research we will provide an introduction to Cloud Computing Systems, P2P Systems, and the advantages as well as the limitations of both configurations. We then describe the concept of a Cloud-Based P2P System, its basic architecture, and its possible implementations. We also describe the possible application of a Cloud-Based P2P System as a platform for a multimedia streaming service. Also we will enhanced the algorithms to facilitate streaming in such an application is also described. Finally, we try to solve the NAT problem, problem (iv), by presenting two NAT- aware peer sampling services (PSS): GOZAR and CROUPIER. Traditional gossip-based PSS breaks down, where a high percentage of nodes are be- hind NATs.

1. Introduction

Peer-to-peer overlay networks are one of the efficient methods to stream live contents from a single source to a large number of receivers over the internet. It offers many advantages in terms of high resource utilization, potential scalability and ease of deployment (Pianese et al. 2011). Unfortunately, the real time constraint such as the inception gap between the streams (time between the generation of the chunk and its actual delivery to the user) still exist as a challenging problem. Events that are live streamed tend to be highly time sensitive and Quality of service (QoS) remains an uncompromised issue. Bottlenecks in the available bandwidth, Network fluctuations and dynamic nature of peers may degrade the performance of streaming.

A potential solution for this problem is assisting the P2P streaming network with cloud computing infrastructure to guarantee the desired level of QoS. In P2P cloud, rented resources (computational nodes and storage nodes) from the cloud are added on demand to the overlay to increase the amount of total available bandwidth and the probability of receiving the video on time.

LIVE media streaming over the Internet is getting more popular every day. The conventional solution to provide this service is the client-server model, which allocates servers and network resources to each client request. However, providing a scalable and robust client-server model, such as Youtube, with more than one billion hits per day is very expensive. There are few companies, who can afford to provide such an expensive service at large scale. An alternative solution is to use IP multicast, which is an efficient way to multicast a media stream over a network, but it is not used in practice due to its limited network-level support by Internet Service Providers.

Another approach is to use the application level multicast, which utilizes overlay networks to distribute large-scale media streams to a large number of users (nodes).

In this model, nodes that have all or part of the requested media can forward it to the requesting nodes. Since each node contributes its own resources, the capacity of the whole system grows when the number of nodes increases. Hence, P2P overlays can provide media streaming services at large scale, but with a relatively lower cost for the service provider than that of the client-server model, as network traffic and data storage/processing costs are pushed out to peer nodes.

1.1. Peer To Peer Streaming

Live media streaming over the Internet is getting more popular every day. The conventional solution to provide this service is the client- server model, which allocates servers and network resources to each client request. However, providing a scalable and robust client-server model, such as You Tube, with more than one billion hits per day is very expensive. An alternative solution is to use IP multicast, which is an efficient way to multicast a media stream over a network, but it is not used in practice due to its limited network-level support by Internet Service Providers. Another approach is to use the application level multicast as shown

in Figure 1.1, which utilizes overlay networks to distribute large-scale media streams to a large number of users (nodes). A Peer-to-Peer (P2P) overlay is a type of overlay network in which each node simultaneously functions as both a client and a server. In this model, nodes that have all or part of the requested media can forward it to the requesting nodes. Since each node contributes its own resources, the capacity of the whole system grows when the number of nodes increases. Hence, P2P overlays can provide media streaming services at large scale, but with a relatively lower cost for the service provider than that of the client-server model, as network traffic and data storage/processing costs are pushed out to peer nodes.

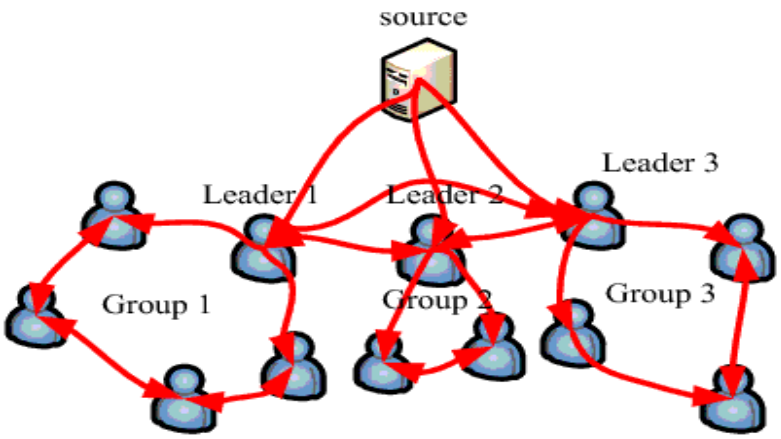

Figure 1 Application Level Multicast Using P2P Overlay

The goal of P2P streaming mechanism is to maximize quality of the peers in a scalable fashion while accommodating the heterogeneity and churn among the participating peers. The question still remains a mystery as how successful is this new trend of technology at providing a good quality of Service (QoS) to end users. The QoS is defined in terms of two metrics in live streaming: Playback Continuity and Playback Latency. To have a high playback continuity, or smooth media playback, nodes should receive data blocks of the stream with respect to certain timing constraints; otherwise, either the quality of the playback is reduced or its continuity is disrupted. Likewise, to have low playback latency, nodes should receive points of the media that are close in time to the most recent part of the media delivered by the provider.

1.2. Cloud Computing For Streaming

The massive scale of Cloud Computing enables popularization of the internet and the growth of multimedia services. Cloud computing provides pay-as-you-go services, infinite scalability and guarantees universal available resources. Cloud computing virtualizes systems by pooling and sharing of resources. Systems and storage can be provisioned as needed from a centralized infrastructure, costs are assessed on a metered basis, multi-tenancy is enabled and resources are scalable with agility (Amir et al. 2012). Cloud renders different services on demand and in this thesis Infrastructure as a Service (IaaS) has been adopted to improve the Quality of Service in multimedia streaming environment (Li et al. 2014). IaaS provides virtual machines, virtual storage, virtual network and other hardware assets as resources that clients can provision. Systems (VM's running applications), storage and network can be virtualized and have sufficient flexibility to provide acceptable distributed WAN application performance (Ab Wahid et al. 2015). Scaling a system can be done by scaling up vertically to more powerful systems or by scaling out horizontally to more in number but less powerful systems. This is a fundamental architecture decision that is affected by the types of workloads that cloud computing systems are being asked to perform.

To scale vertically, resources are added to the system to make it more powerful. For example, in a cloud based system that has dual processor machine, can be replaced with a quad processor machine instance to scale up the resources. Scaling up can be performed by adding more memory, more network throughput and other resources to a single node. Vertical scaling uses virtual system to run more virtual machines, more daemons on the same machine instance or take advantage of more RAM and faster computer times. Horizontal scaling or scale out adds capacity to a system by adding more individual nodes (Amoretti et al. 2015). In a system having dual processor machine instance, the system can be scaled out by adding more machines of the same type or other types of commodity system. Horizontal scaling runs distributed applications more efficiently and is effective in using hardware more efficiently due to the ease of aggregating and partitioning of resources.

1.3. Cloud Assisted Multimedia Streaming

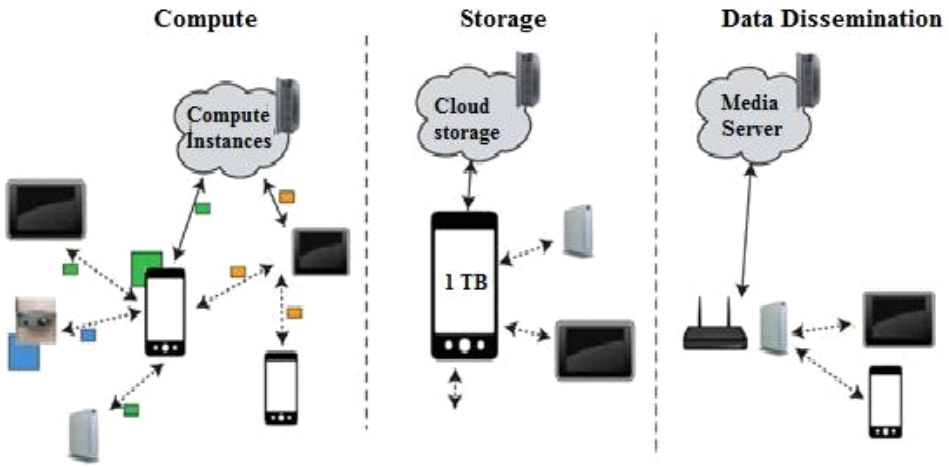

Figure 2 Role of Cloud in Multimedia Streaming

Multimedia Streaming over the Internet has become one of the major broadcast transmissions over the last few years. Many trends have come together to make this transition into reality such as broadband networks, high-capacity commodity disks, low cost computing power and now recently Cloud computing (Bin et al. 2014). Video streaming is one of the technologies that benefit greatly on deploying in the cloud. Many of these services already have a cloud based infrastructure, where some of them are content servers, others are systems for caching/storing the content and some services can be used to modify content in an appropriate format and with many other additional services besides. Streaming media files is the generation of chunks by the media server and played back by a client assisted with data dissemination. Streamed material can be live or on-demand. Live streaming is otherwise known as progressive streaming or progressive download, while on-demand streaming is from material that is already stored onto the disk (Feng et al. 2012).

2. Cloud Assisted Peer to Peer Streaming

Peer to peer live streaming is a popular technology used in many academic and commercial live streaming applications (Pianese et al. 2011). In such systems, one of the most important challenges is to guarantee Quality of Service in spite of the fluctuations in

the network and dynamic behavior of the peers (Baldesi et al. 2014). High Playback continuity and minimum playback delay are the two important parameters that need attention when considering live streaming events. One interesting methodology to solve this issue is Cloud assisted peer to peer streaming system. A number of helpers can be rented from the cloud on demand basis which is required to guarantee content propagation (Amir Payberah et al. 2012). A helper could be a Computation node used for increasing the computation power of the system or can be a storage node which is used for backup purpose. The computational nodes (virtual machines) increase the total available upload bandwidth, thus, potentially reduces the playback delay.

2.1 Cloud based Transcoding

Cloud computing consists of a cluster of distributed computers where each of them are engaged in their tasks independently and all of them together provide a parallel computing where execution time of tasks is minimal. Since the computers are heterogeneous, cloud computing is extendable and relatively inexpensive. Map/Reduce is a distributing cloud computing model (Seungho et al. 2014). As showed in Figure 1.11, the input data would be divided into many splits which are submitted as sub-tasks and handled by mappers in multiple computers. The output of mappers would be handled by reducers to provide the final output of the reduced video in the required format.

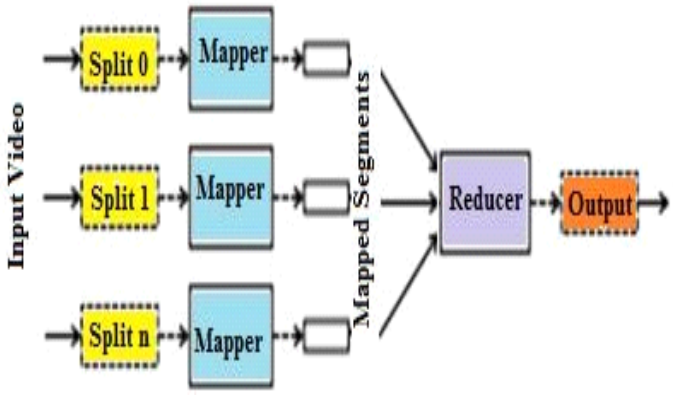

Figure 1.11 MapReduce Based Transcoding

Map-Reduce-based cloud computing focuses on the scheduling for sub-tasks to multiple computers (Powers Keke et al. 2014). The sub-tasks are launched in parallel with processing results concatenated to the final output sequences.

2.2 Literature Review

Cloud based live streaming has become one of the prominent area of research in recent years. Past few years researchers had carried out extensive work in peer to peer multimedia streaming (Ab et al. 2015).This chapter deals with the study of existing approaches related to live streaming and approaches that improved Quality of Service (Baldesi et al. 2014). Detailed survey has been carried out on existing P2P overlay topologies for live streaming, Content dissemination algorithms, Incentive mechanism to motivate the peers in the network, Network Address Translation for host identification residing behind firewall, Cloud based live streaming, Bandwidth sharing mechanism, Resource allocation strategies and Cloud based Transcoding.

Fei Chen et al. (2015) presented a generic framework that facilitates cost-effective cloud service for crowd sourced live streaming. Through adaptive leasing, the cloud servers were provisioned in a fine granularity to accommodate geo-distributed video crowdsourcers. An optimal solution was proposed to deal with service migration among cloud instances with diverse lease prices. The approach was said to be further enhanced with crowdsourcer prediction through user behavior analysis from real-world measurements.

Sung-Yen Chang et al. (2012) proposed a dynamic and adjustable multimedia streaming service architecture over cloud computing. It adopted a device profile manager concept which made use of the information concerned with the streaming parameters, device specification and network data. The system was evaluated in terms of power consumption based on the configuration of the system.

Mehdi Seydali Seyfabad & Behzad Akbari (2014) proposed CAC-Live, a P2P live video streaming system that used cloud to overcome the challenges caused due to insufficiency of resources. The system was built based on centralized algorithms for monitoring the QoS perceived by the peers. These algorithms were designed to manage the virtual machines that are added to or removed from the network. At every iteration in the system the peer

periodically sends information about its received QoS in terms of playback latency and based on this information, the system decide whether to add or remove virtual machines obtained from the cloud.

Korpeoglu et al. (2013) proposed 'Dragonfly Architecture' where, Multipoint streaming applications were assisted using Cloud resources. Applications are designed based on Multi-user interaction and this leads to communication between users from various geographical locations. The model contributed location aware 3-Tier architecture for live streaming. The peers constitute the bottom layer where, the geo-dispersed users form an overlay, Edge cloud forms the middle layer which handles local cluster of users and the management layer forms the top most layer which handles multiple streaming session.

Zhang Yi et al. (2015) proposed a bandwidth allocation mechanism where the arrival rate of peers was considered as Poisson distribution process. The performance of this mechanism was proved by mathematical analysis. The procedure involved in keeping track of the events that occurred at a given time interval. The objective of this system was to achieve a steady state swarm until the chunks are downloaded by all peers. The bandwidth allocation mechanism proved 93% of performance while efficiently allocating the bandwidth among the total number of peers. The method has not been tested for dynamic scenarios.

Babaglou, Marzolla, and Tamburini show us the benefits of utilizing a Cloud-Based P2P System instead of other system models. However, there is no reason to implement such a complicated system unless there is a definite and tangible application. Streaming of high quality audio and video has always been a challenge due to storage and bandwidth constraints. We can see that the use of a Cloud-Based P2P System is ideally suited for this situation.

CLive is one of the applications which attempts to use the advantages of Cloud-Based P2P Systems to provide a video streaming service with high QoS. Another existing application is the Novasky service which has been deployed in Tsinghua University. Novasky provides bit rates of 1-2 Mbps while managing over 1000 high quality video streams. This gave us the idea of determining an algorithm to find the nearest peer which is capable of streaming the desired content, thereby decreasing latency even further. While research has been done on

various other system models and architectures, we have devised a model that we believe should ideally suit the needs of the client while ensuring system manageability. Such a system model can be utilized by both small scale and very large scale users, as the system allows the interconnection between both types of systems. The utility of a Cloud-Based P2P System was demonstrated by Ke Xu et al., with the virtues of decentralization being clearly described. The next step in this application's development would involve accessing the Cloud-Based P2P System via a mobile device, such as a smartphone. This has been analyzed thoroughly in the development of Stream Smart.

Table 1 Comparison of the Existing Methodologies

S.NO	AUTHOUR	APPROACH	PROS	CONS
PEER TO PEER LIVE STREAMING				
1.	Miao Wang et al. (2013)	Linear programming models	Efficient bandwidth allocation for cross channels.	Lacks ability in analyzing the Performance under dynamism.
2.	Hamed Azarpira et al. (2013)	Optimal Structured overlay	Efficient and stabilized data dissemination. Cost efficient bandwidth allocation.	Lacks Robustness in churnscenarios.
3.	Haiying Shen (2014)	Social-network-aided efficient P2P live streaming	Supports Multi- channel streaming. Low switch delay and low server overhead.	Maintenance cost is very highcomplexity exist in communication between nodes
4.	Matthias Wichtlhuber (2014)	Transition-enabled P2P streaming	Highly resilient during flash-crowdscenarios.	Time complexity is high during Multi-Tree configuration with O((log n)).
5.	Di Wu et al. (2013)	Sub gradient method	Both node heterogeneity and peerchurn was considered while Constructing an overlay.	Fairness optimization is notconcentrated

6.	Chun-Yuan et al. (2014)	Swarm-based P2P live streaming	Bridges the gap between rate-distortion analysis of compressedvideo.	No chunk identification strategy in each node which leads in missing packets.
7.	Yishuai Chen et al. (2015)	Fluid model	Highly elastic and scalable. Capable of recovering during flash crowd scenario.	Communication overhead andlink failure occurs frequently.
8.	Nazanin et al. (2015)	ISP-friendly P2P streaming	Efficiently detects missing blocks with a minimum coordination overhead.	Not prone to globalized scenario.

3. Problem Statement

Peer to Peer live streaming overlays are dynamic in nature where nodes join/leave/fail continuously which collapses the overlay structure. There occurs a tradeoff between high playback continuity and minimum playback delay; however it is possible to increase the playback continuity by adopting large streaming buffers which results in maximum delay. Improving playback delay requires high bandwidth inside the overlay, as well as at the media source. Media source provided with high bandwidth does not solve the problem where, disruption occurs inside the overlay which affects the continuity of streams. The problem consists in selecting the right type of helpers from the cloud. Provisioning too few helpers could be impossible to achieve the desired level of QoS.

We consider a network consisting of a dynamic collection of nodes that communicate through message exchanges. Nodes could be peers, i.e., edge computers belonging to users watching the video stream, helpers, i.e., computational and storage resources rented from an IaaS cloud, and the media source that generates the video stream and starts its dissemination towards peers.

Each peer is uniquely identified by an ID, e.g., composed by an IP address and a port, required to communicate with it. We use the term swarm to refer to the collection of all peers. The swarm forms an overlay network, meaning that each peer connects to a subset of nodes in the swarm (called neighbors). The swarm is highly dynamic: new peers may join at any time, and existing peers may voluntarily leave or crash. Byzantine behavior is not considered in this work

There are two types of helpers: (i) an active helper (AH), which is an autonomous virtual machine composed of one or more computing cores, volatile memory and permanent storage, e.g., Amazon EC2, and (ii) a passive helper (PH), which is a simple storage service that can be used to store (PUT) and retrieve (GET) arbitrary pieces of data, e.g., Amazon S3. We assume that customers of the cloud service are required to pay for computing time and bandwidth in the case of AHs, and for storage space, bandwidth and the number of PUT/GET requests in the case of PHs. This model follows the Amazon's pricing model [89, 90].

4. Objectives

The objective of this research is to understand multimedia live streaming infrastructure, applications, and features fully. Also, it aims to identify and analyze the p2p live streaming, that affect cloud computing performance and understand the techniques P2P multimedia live streaming and hybrid P2P used to enhance the quality of the services in the cloud. The main objective of this research is to make improved QoS in cloud environments.

- **To provide** an overview of multimedia live streaming, applications, and features.
- **To provide** an overview of cloud assisted peer to peer streaming.
- **To identify** the suitable techniques, components and the requirement for multimedia live streaming.
- **To improve** the Quality of Service in Live streaming environments (Streaming live media with a high QoS, i.e., high playback continuity, and low playback latency, over a P2P overlay.
- **A cloud-assisted P2P live streaming system**, CLIVE, that guarantees a higher bound on playback latency, if there exists bottlenecks in the available upload bandwidth inside the P2P overlay, by renting cloud resources

5. Research Methodology

In order to achieve the goal of this research, I have chosen the design of science methodology because my research focuses on the design and improvement technologies for enhance the quality of the services in the cloud model.

References

1. Ab Wahid, N, WB, Jenni, K, Mandala, S & Supriyanto, E 2015, 'Review on Cloud Computing Application in P2P Video Streaming', Big Data, Cloud and Computing Challenges: Procedia Computer Science, Elsevier, Chennai, India, Vol.1, no.1, pp. 185-190.
2. Amoretti, M, Grazioli, A & Zanichelli, F 2016, 'An autonomic approach for P2P/cloud collaborative environments', Peer-to-PeerNetworking and Applications, Vol. 9, no. 6, pp. 1-16.
3. Andriopoulou, FG, Birkos, K & Lymberopoulos, D 2015, 'P2Care: A dynamic peer-to-peer network for collaboration in personalized healthcare service delivery', Computers in Industry, Vol. 69, no.1,pp. 45-60.

4. Bellavista, P, Reale, A, Corradi, A& Koutalas, S 2014, 'Adaptive fault- tolerance for dynamic resource provisioning in distributed stream processing systems', International conference on Extended Database Technology: proceedings of EDBT/ICDT, Athens, Greece, Vol. 3, no. 6, pp. 327-334.

5. Baldesi, L, Maccari, L & Cigno, L 2014, 'Improving p2p streaming in community-lab through local strategies', International conference on wireless and mobile computing, networking and communications: proceedings of Wi Mob '14, Larnaca, Cyprus, Vol. 1, no. 1, pp. 33- 39.

6. Bianchi, G, Melazzi, NB, Bracciale, L, Piccolo, FL, & Salsano, S 2010, 'Streamline: An Optimal Distribution Algorithm for Peer-to-Peer Real-Time Streaming', IEEE Transactions on Parallel and Distributed Computing, Vol. 21, no. 6, pp. 857-871.
7. Bin Cheng& Media Paa, S 2014, 'MediaPaaS: A Cloud-based Media Processing Platform for Elastic Live Broadcasting', International Conference on Cloud Computing: proceedings of IEEE CloudComputing Anchorage, Alaska, Vol. 1, no. 1, pp. 713-720.
8. Deltouzos, K, Gkortsilas, I, Efthymiopoulos, N & Denazis, S 2015, 'Liquidstream II—Scalable P2P overlay optimization with adaptive minimal server assistance for stable and efficient video on demand', Peer-to-Peer Networking and Applications, Vol. 8, no. 2, pp. 260-275.

9. Detti, A, Ricci, B & Blefari-Melazzi, N 2015, 'Mobile peer-to-peer video streaming over information-centric networks', Computer Networks, Vol. 81, pp. 272-288.

10. Feng Wang, Liu, J, Chen, M & Wang, H 2015, 'Migration towards Cloud-Assisted Live Media Streaming', IEEE/ACM Transactions on Networking, Vol. 2, no. 9, pp. 1-9.
11. Ghimire, J, Mani, M, Crespi, N & Sanguankotchakorn, T 2015, 'Delay and capacity analysis of structured P2P overlay for lookup service', Telecommunication Systems, Vol. 58, no. 1, pp. 33-54.

12. Preetha Evangeline, D & Anandhakumar, P 2015, 'SEASONS-A Scalable 4-Tier Cloud Architecture for Handling Heterogeneity of Mobile Devices in Live Multimedia Streaming', ARPN Journal ofEngineering and Applied Sciences, Vol. 10, no. 5, pp. 2017- 2022.

13. Radu Tudoran, Olivier Nano, Ivo Santos, Alexandru Costan, Hakan Soncu, Luc Bouge& Gabriel Antoniu 2015, 'JetStream: Enabling high throughput live event streaming on multi-site clouds', Future Generation Computer Systems, Elsevier, Vol. 54, no.1, pp.677-683.

Researcher Student Signature Guide Signature Director Signature